2004

AI on the GPU

Darken, Christian J.

Extended abstract for poster presentation at the ACM Workshop on General Purpose Computing on Graphics Processors and SIGGRAPH, 2004.
http://hdl.handle.net/10945/36612

AI on the GPU

Christian J. Darken, E. Ray Pursel, J. Steve Correia
MOVES Institute, Naval Postgraduate School
Monterey, CA 93943
cjdarken@nps.edu

A major AI challenge for entity-level military simulations (i.e. simulations of individual soldiers, vehicles, etc.) and computer games is efficiently determining which entities on a virtual battlefield can see each other [1]. Entity inter-visibility is traditionally modeled by tracing a line of sight (LOS), i.e. entity *A* sees entity *B* if the line segment between *A*'s eyes and the top of *B* is unobstructed. This model is, however, highly inaccurate. Computing lines of sight to multiple points on the target entity improves accuracy, but at the cost of further loading the CPU.

A substantial improvement would be to base the entity inter-visibility calculation on the visible fraction of the entity's surface area [4]. This idea has a pedigree going back to the 1950's, and a lot of military simulation work and hardware development has made use of it. A CPU-based version of this idea has been used in at least one military simulation [5], though not necessarily on models with complex geometry in real time.

An even more significant improvement over traditional LOS would be to base the visibility determination on the same type of information provided to a human player, i.e. the actual pixel values of *B* as seen from *A*'s point of view. This class of inter-visibility algorithms is clearly related to computer vision [3] while being fundamentally different in that the ground truth is known and can be used in the computation. We propose to make this significantly more complex approach to inter-visibility determination feasible by doing much of the computation on the GPU.

To determine whether *A* sees *B*, our approach begins by performing a low resolution render-to-texture of a small window centered on *B* from *A*'s viewpoint. A second render of the same window using false-color to code object identities [2], to mention one specific technique, is used to determine which pixels are part of *B* (the "figure") and which are not (the "ground"). Color and textural differences between the figure and ground can now be calculated based on computer vision algorithms. The results should allow an estimation of whether *A* sees *B* of unprecedented sophistication. For the first time, it may be possible to realistically account for such factors as shadow, camouflage, silhouetting, and smoke in real time. Much or all of this computation can be done on the GPU.

An inventory of image characteristics that we are considering for our inter-visibility models, from simple to complex, is given below.

- Visible surface area
- Brightness differences
- Color histogram differences
- Textural differences
- Salience of characteristic edges
- Visibility of object sub-components

References

[1] Darken, C. "Visibility and Concealment Algorithms for 3D Simulations", *Proceedings of Behavior Representation in Modeling and Simulation (BRIMS) 2004*.

[2] Chover, M., Hagen, H., and Tost, D. "Parallel Visibility Test and Occlusion Culling in Avango Virtual Environment Framework", *Proceedings of Eurographics 2003*.

[3] Fung, J. and Mann, S. "Computer Vision Signal Processing on Graphics Processing Units", *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004)*, Montreal, Quebec, Canada, May 17--21, 2004.

[4] Reece, D. and Wirthlin, R. (1996) "Detection Models for Computer Generated Individual Combatants", 6th Conference on Computer Generated Forces and Behavioral Representation, University of Central Florida

[5] TASC and SAIC (1996) LibCTDB User Manual and Report. The Analytical Science Company and Science Applications International Corporation.